APÉNDICE C. COMPARACIÓN ENTRE EL LENGUAJE DEL SISTEMA MONOLÍNEA Y EL LENGUAJE DEL SISTEMA MULTILÍNEA

C.1 INTRODUCCIÓN

El objetivo de este apéndice es comentar las diferencias más significativas entre el lenguaje de que estaba dotado el anterior sistema monolínea y el lenguaje que se utiliza en el nuevo sistema multilínea, con el fin de facilitar la adaptación de las aplicaciones ya desarrolladas al nuevo sistema multilínea.

Para ello, veremos un ejemplo de una aplicación completa, que cubre casi todas las diferencias. Al final comentaremos otros aspectos que no quedan cubiertos con el ejemplo.

C.2 DESCRIPCIÓN DEL SERVICIO DE BUZÓN TELEFÓNICO

C.2.1 INTRODUCCIÓN

Este servicio permitirá a cualquier persona que éste dada de alta en él recibir mensajes de un buzón telefónico privado y depositar mensajes a otros usuarios del servicio. Las personas que no estén dadas de alta podrán depositar mensajes en el buzón con destino a usuarios registrados del mismo.

Si se desea depositar un mensaje en el buzón, bastará con indicar el código del destinatario. Si se quiere recibir los mensajes habrá que indicar el código personal y una CLAVE secreta de acceso. El servicio proporcionará todos los mensajes disponibles, empezando por aquellos que se han recibido desde la última llamada al mismo.

El sistema soporta tanto reconocimiento de voz como detección de tonos multifrecuencia (DTMF), maneja un vocabulario compuesto por los dígitos del *cero* al *nueve* y las palabras *si*, *no*, *ayuda* y *cancelar*.

Para enviar mensajes al usuario se utiliza la reproducción de mensajes pregrabados, por su mayor calidad, siempre que esto sea posible. También se utiliza la conversión texto-habla para sintetizar palabras cortas que dependen del usuario como, por ejemplo, su nombre.

C.2.2 FUNCIONAMIENTO

Cuando el sistema detecta una llamada descuelga y envía un mensaje de bienvenida al usuario. Posteriormente se le da la oportunidad de escuchar unos mensajes de ayuda sobre el funcionamiento del sistema.

A continuación se presenta el menú principal, con las opciones de: dejar un mensaje, escuchar los mensajes recibidos o salir de la aplicación.

Si el usuario quiere dejar un mensaje en el buzón privado de otro usuario del servicio, se le pedirá el código del destinatario y el sistema comprobará que está dado de alta en el servicio. Si es así, indicará al usuario que deje su mensaje después del pitido, lo grabará y volverá al menú principal. En caso de que el destinatario no tenga acceso al servicio, se informará al usuario de ello y volverá al menú principal.

Si el usuario decide escuchar los mensajes de su buzón, el servicio le solicitará el código personal y su clave. Tras comprobar que tiene acceso al servicio (en caso contrario, le informa y le envía al menú principal) le reproduce todos los mensajes que se han recibido desde su última llamada y se presenta un menú secundario con las siguientes opciones:

- Cambiar la clave.
- Escuchar todos los mensajes de su buzón y, tras cada mensaje, indicar si desea borrarlo o no.
- Salir de la aplicación.

Si decide salir de la aplicación se le reproduce un mensaje de despedida y se cuelga, quedando el sistema en espera de una nueva llamada.

El diagrama de flujo de la aplicación es el siguiente:

C.3 DIFERENCIAS ENTRE AMBOS LENGUAJES

C.3.1 ASPECTOS GENERALES

- Algunas de las variables internas ya no están disponibles y las demás han cambiado de nombre. Ver *Apéndice A Manual del Usuario*, para más información.

- Las variables están tipadas, y su nombre lo refleja , ***n_*** para variables numéricas y ***s_*** para variables cadena de texto.

- Es posible incluir un fichero externo con la instrucción ***#include*** en cualquier punto de la aplicación (debe tener la extensión PRO). Ver *#include "verifcad"* al comienzo de la sección de subrutinas. El contenido del fichero "verifcad.pro" se puede ver en el *Apéndice A Manual del Usuario*.

- Se permiten los comentarios. Deben comenzar con /* y acabar con */. No son anidables; es decir, dentro de un comentario no podemos poner otro comentario.

C.3.2 SECCIÓN ERRORES

- Los nombres de los tratamientos de error no van entre comillas.

- No existe la instrucción ***reiniciar***. Se debe sustituir por un ***goto*** a una etiqueta de la sección aplicación, previa a la intrucción ***esperar_llamada***.

C.3.3 SECCIÓN SUBRUTINAS

- La última instrucción de una subrutina debe ser necesariamente ***retornar***.

- Es posible incluir llamadas a subrutinas definidas posteriormente.

- Debido a que no es posible anidar instrucciones del tipo ***elegir_opcion*** ha sido necesario definir una nueva subrutina, llamada *sub1*. De esa subrutina no salimos normalmente (con la instrucción ***retornar***), sino que saltamos (con la instrucción ***goto***) a etiquetas dentro de la *SECCION_APLICACION*. Por tanto, es necesario utilizar la *función interna* ***RESET_GOSUB*** para vaciar la pila donde se guardan los retornos de subrutina.

C.3.4 SECCIÓN APLICACIÓN

- Las llamadas a subrutinas deben incluir los dos puntos después del nombre de la subrutina.

- Cualquier referencia a una etiqueta debe incluir los dos puntos.

- Los argumentos de las funciones internas deben ir entre paréntesis.

- Para cambiar el modo de reconocimiento, en lugar de utilizar la instrucción ***set_reconocimiento***, hay que asignar a la variable ***s_tipo_reco*** los valores 'voz', 'DTMF' o 'ambos', según se desee reconocimiento mediante voz, detección de tonos multifrecuencia o ambos, respectivamente.

- La variable ***n_timeout_sistema***, que no estaba presente en el sistema monolínea, indica la duración máxima de una llamada, en minutos. Es una medida de seguridad, y si se supera, el sistema colgará la llamada en curso, estando así en disposición de atender una nueva llamada.

- La función interna ***Montar_Query*** no está disponible, pero puede usarse la función predefinida ***STRCAT***, que realiza la misma función.

- La instrucción ***elegir_opcion*** requiere el término ***switch*** antes de detallar los case, cuyos valores deben ir siempre entre comillas.

- La instrucción ***reconocer_cadena*** debe incluir dos ***asignar_a***, pues en la variable que indique el segundo, se guardará el segundo candidato de la cadena reconocida.

- La función ***Segunda_Vez*** no existe. En su lugar se utiliza una variable numérica y se decide el flujo de la aplicación en función de su valor con la instrucción condicional ***if***.

- No es necesario incluir la extensión del fichero en la instrucción reproducir, aunque se puede hacer. Si no se incluye se asume la extensión adecuada según el hardware presente. Ver *Apéndice A Manual del Usuario*, para más información.

- Las funciones internas **IDAPI** se han sustituido por **ODBC**, aunque IDAPI sigue estando presente. Su funcionalidad es similar. Todas las instrucciones referentes a ODBC incluyen al menos dos parámetros: el nombre de la base de datos (*buzon*, en este caso), y el tipo de la base de datos (*access*, en este caso). El primero de ellos debe ser un origen de datos ODBC válido, que debe figurar en el Administrador de ODBC de Windows (se encuentra en el Panel de Control). El segundo es el tipo de base de datos (access, dBase, etc) y se ha mantenido por compatibilidad con IDAPI, pero en realidad no se utiliza, pues viene determinado por el primer parámetro. Ver el *Capítulo 7* para más información sobre ambos métodos de acceso a bases de datos locales.

- La intrucción ***Verificar_Cadena*** ha desaparecido. En su lugar **existe una subrutina**, llamada ***VerifCadena***, que se incluye como fichero aparte en la sección de subrutinas, que se encarga de verificar cualquier cadena que se le pase, y haciendo uso de la cadena con los segundos candidatos de cada palabra que componen la cadena a verificar. Para usarla, basta colocar la variable a verificar y su segundo candidato en las variables *s_cadena* y *s_segundo*, respectivamente y llamarla. Después hay que volver a realizar las asignaciones anteriores, pero en sentido contrario.

  Ejemplo:

  En la sección subrutinas, se debe incluir:

```
SECCION_SUBRUTINAS:
#include "verifcad"          /* el fichero verifcad.pro debe existir */
```

Y si queremos verificar la cadena *s_codigo*, y suponemos que la cadena con los segundos candidatos se llama *s_codigo_segundo*, haremos:

```
s_cadena = s_codigo;
s_segundo = s_codigo_segundo;
gosub VerifCadena:;
s_codigo = s_cadena;
s_codigo_segundo = s_segundo;
```

Es necesario recoger la cadena con los segundos candidatos, pues en la subrutina **VerifCadena** se puede llegar a solicitar al usuario que repita un dígito, momento en el que se generará un nuevo segundo candidato, que lo necesitaremos si hemos de hacer otra verificación posterior sobre la misma cadena. Si el tipo de reconocimiento es **DTMF** (*s_tipo_reco = "DTMF"*), no se utilizan los segundos candidatos.

C.3.5 OTRAS CONSIDERACIONES

- En el ejemplo no se muestra, pero para **redirigir llamadas** entre centralitas, tanto ALCATEL como IBERCOM, se ha desarrollado otra **subrutina** de alto nivel, llamada ***Redirigir***. Para utilizarla hay que colocar en la variable *s_central* el tipo de central y en la variable *s_numero* el número al cual queremos redirigir la llamada.

  Por **ejemplo**: para redirigir la llamada que estamos atendiendo al número 6824 de Ibercom, basta hacer:

```
s_central = "IBERCOM";
s_numero = "6824";
gosub Redirigir:;
```

C.3.6 COMPARACIÓN ENTRE BUZON.LDD Y BUZON.PRO

BUZON.LDD

```
SECCION_ERRORES:

    Tratamiento "no_reconocido":
            sintetizar "Esta opción no está activa en este momento.";
            reintentar;
    Fin_Tratamiento;


    Tratamiento "longitud_corta":
            reproducir corto.mst;
            reintentar;
    Fin_Tratamiento;


    Tratamiento "colgado":
            colgar;
            reiniciar;
    Fin_Tratamiento;


    Tratamiento "fin_intentos":
            sintetizar "Se le han acabado las posibilidades. Hasta la próxima
            vez.";
            colgar;
            reiniciar;
    Fin_Tratamiento;
```

BUZON.PRO

```
SECCION_ERRORES:

TRATAMIENTO NO_RECONOCIDO
        sintetizar("Esta opción no está activa en este momento. Vuelva a
        intentarlo.”;);
        reintentar;
FIN_TRATAMIENTO


TRATAMIENTO LONGITUD_CORTA
        reproducir( "corto";);
        reintentar;
FIN_TRATAMIENTO


TRATAMIENTO COLGADO
        colgar(;);
        goto ESPERAR: ;
FIN_TRATAMIENTO


TRATAMIENTO FIN_INTENTOS
        sintetizar("Se le han acabado las posibilidades. Hasta la próxima vez.";);
        colgar(;);
        goto ESPERAR: ;
FIN_TRATAMIENTO
```

```
Tratamiento "timeout":
        reintentar;
Fin_Tratamiento;
```

```
TRATAMIENTO TIMEOUT
/*
    si se reconoce una palabra (instrucción elegir_opcion) se reproduce nada
    si se reconoce una cadena (instrucción reconocer_cadena) se reproduce rep_dig
*/
s_reco_estado = s_estado_reco;      /* las variables internas no pueden participar */
s_reco_tipo = s_tipo_reco;          /* en una condición de la instrucción if       */

if (s_reco_tipo == "voz")
        if (s_reco_estado == "palabra")
                reproducir("nada";);        /* Hable un poco más alto... */
        endif
        if (s_reco_estado == "cadena")
                reproducir("rep_dig";);     /* Repita el último dígito... */
        endif
endif
if (s_reco_tipo == "ambos")
        if (s_reco_estado == "palabra")
                reproducir("nada";);
        endif
        if (s_reco_estado == "cadena")
                reproducir("rep_dig";);
        endif
endif
if (s_reco_tipo == "DTMF")
        if (s_reco_estado == "cadena")
                reproducir("rep_dig";);
        endif
endif
        reintentar;
FIN_TRATAMIENTO
```

```
SECCION_SUBRUTINAS:


    Subrutina AYUDA:
            interrupcion = 0;
            sintetizar help.txt;
            interrupcion = 1;
    Fin_Subrutina;



    Subrutina FALLO_SISTEMA:
          sintetizar "Fallo en el sistema. No puedo atenderle ahora. Perdone las
          molestias que esto pueda ocasionarle.";
          colgar;
          salir;
    Fin_Subrutina;
```

```
SECCION_SUBRUTINASs:
#include "verifcad"

SUBRUTINA AYUDA:
          n_interrumpir = 0;
          sintetizar ("Para realizar una elección entre las posibles opciones que se le
          presenten...";);
                    /*  texto contenido en el fichero 'help.txt' */
          n_interrumpir = 1;
          retornar;
FIN_SUBRUTINA

SUBRUTINA FALLO_SISTEMA:
          sintetizar("Fallo en el sistema. No puedo atenderle ahora. Perdone las molestias
          que esto pueda ocasionarle.";);
          colgar(;);
          goto ESPERAR: ;
FIN_SUBRUTINA
```

```
Subrutina ALMACENAR:
        pitar;
        grabar mensaje5.mst con_duracion 5;
        pitar;
        funcion_interna Almacenar_En_Disco
        ("mensaje5.mst",CODIGO;fichero);
                case -1:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;


        funcion_interna Montar_Query("5","insert into mensaje (
        \"codigo\" , \"fichero\" , \"escuchado\" ) values
        ('",CODIGO,"','",fichero,"','N')";query);
                case -1:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;

        funcion_interna Buscar_Elemento("buzon","access",query);
                case -2:
                case -5:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;
Fin_Subrutina;
```

```
SUBRUTINA ALMACENAR:
        s_anterior = s_DIR_GRABACION;
        SPRINTF(s_dir,"%s%s",s_anterior,s_codigo;);
        MKDIR(s_dir;);
        MKTEMP(s_dir;s_fichero0);
        STRCAT(s_dir,"\\";s_dir);
        s_DIR_GRABACION = s_dir;
        pitar(;);
        GRABAR_FICHERO(s_fichero0,"5","duracion";);
        pitar(;);
        s_DIR_GRABACION = s_anterior;
        /* en la base de datos hay que guardar \directorio\fichero */
        SPRINTF(s_fichero00,"\%s\%s",s_codigo,s_fichero0;);

        s_trozo1="insert into mensaje ( \"codigo\" , \"fichero\" ,
        \"escuchado\" ) values ('";
        STRCAT(s_trozo1,s_codigo,"','",s_fichero00," ','N')";s_query);

        ODBC_BUSCAR("buzon","access",s_query;);
        funcion_interna RESULTADO_ANTERIOR(;)
        switch
                case "-2" :
                case "-5" :
                        gosub FALLO_SISTEMA:;
                        break;
        fin_funcion_interna;
        retornar;
FIN_SUBRUTINA
```

```
Subrutina ELIMINAR:
        funcion_interna Eliminar_De_Disco (fichero);
                case 0:
                        break;
                case -1:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;

        funcion_interna Montar_Query ("3","delete from mensaje where
        \"fichero\" = '",fichero,"';query);
                case -1:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;

        funcion_interna Buscar_Elemento("buzon","access",query);
                case  1:
                case  0:
                        break;
                case -2:
                case -5:
                        gosub FALLO_SISTEMA;
        fin_funcion_interna;
        sintetizar "Mensaje borrado";
  Fin_Subrutina;

Subrutina BORRAR:
        elegir_opcion
                opciones_activas="SN";
                sintetizar "¿Desea borrar el mensaje?";
                case "SI":
                        gosub ELIMINAR;
                        break;
                case "NO":
                        break;
        fin_elegir_opcion;
  Fin_Subrutina;
```

```
SUBRUTINA ELIMINAR:
        DELETE(s_fichero;);

        STRCAT("delete from mensaje where \"fichero\" =
        '",s_fichero,"';s_query);

        ODBC_BUSCAR("buzon","access",s_query;);
        funcion_interna RESULTADO_ANTERIOR(;)
        switch
                case  "1":
                case  "0":
                        break;
                case "-2" :
                case "-5" :
                        gosub FALLO_SISTEMA:;
                        break;
        fin_funcion_interna;
        sintetizar("Mensaje borrado.";);
        retornar;
FIN_SUBRUTINA

SUBRUTINA BORRAR:
        elegir_opcion
        opciones_activas="SN";
        sintetizar("¿ Desea borrar el mensaje ?.";);
                switch
                case "SI":
                        gosub eliminar: ;
                        break;
                case "NO":
                        break;
        fin_elegir_opcion;
        retornar;
FIN_SUBRUTINA
```

```
Subrutina CAMBIO_CLAVE:
        funcion_interna Montar_Query("7","update usuarios set \"clave\" =
        '",CLAVE_NEW,'" where \"codigo\" = '",CODIGO,'" and \"clave\" =
        '",CLAVE,"';query);
        fin_funcion_interna;

        funcion_interna Buscar_Elemento("buzon","access",query);
                case 1:
                case 0:
                        break;
                case -2:
                case -5:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;
Fin_subrutina;

Subrutina PEDIR_CODIGO:
        reconocer_cadena
                1,8 digitos;
                asignar_a CODIGO;
                sintetizar "Indique el código personal del destinatario.";
        fin_reconocer_cadena;
fin_subrutina;

Subrutina PEDIR_CLAVE:
        reconocer_cadena
                1,8 digitos;
                asignar_a CLAVE;
                sintetizar "Indique su clave secreta.";
        fin_reconocer_cadena;
fin_subrutina;
```

```
SUBRUTINA CAMBIO_CLAVE:
        s_trozo1="update usuarios set \"clave\" = '";
        s_trozo2="' where \"codigo\" = '";
        STRCAT(s_trozo1,s_clave_new,s_trozo2,s_codigo,"' and \"clave\" =
        '",s_clave,"';s_query);

        ODBC_BUSCAR("buzon","access",s_query;);
        funcion_interna RESULTADO_ANTERIOR(;)
        switch
                case  "1":
                case  "0":
                        break;
                case "-2" :
                case "-5" :
                        gosub FALLO_SISTEMA:;
                        break;
        fin_funcion_interna;
        retornar;
FIN_SUBRUTINA

SUBRUTINA PEDIR_CODIGO:
        reconocer_cadena
                1,8 digitos;
                asignar_a s_codigo;
                asignar_a s_codigo_segundo;
                sintetizar("Indique el código personal del destinatario.";);
        fin_reconocer_cadena;
        retornar;
FIN_SUBRUTINA

SUBRUTINA PEDIR_CLAVE:
        reconocer_cadena
                1,8 digitos;
                asignar_a s_clave;
                asignar_a s_clave_segundo;
                sintetizar("Indique su clave secreta.";);
        fin_reconocer_cadena;
        retornar;
FIN_SUBRUTINA
```

```
subrutina AYUDA_VERIFICACION:
        reproducir help3.mst;
        reproducir help4.mst;
        reproducir help5.mst;
 fin_subrutina;
```

```
SUBRUTINA AYUDA_VERIFICACION:
        reproducir ("help3";);
        reproducir ("help4";);
        reproducir ("help5";);
        retornar;
FIN_SUBRUTINA

/*
   Nueva subrutina debido a que no es posible anidar dos instrucciones del tipo
   'elegir_opcion'
*/
SUBRUTINA SUB1:
        elegir_opcion
        opciones_activas = "SN";
        sintetizar("¿ Quiere volver a intentar introducir la clave?";);
        switch
                case "si" :
                        /* como salimos de la subrutina mediante un goto, es
                        necesario vaciar la pila donde se guardan los retornos de
                        subrutina */
                        funcion_interna RESET_GOSUB(;)
                        fin_funcion_interna;
                        goto CAMBIAR_CLAVE: ;
                case "no" :
                        /* como salimos de la subrutina mediante un goto, es
                        necesario vaciar la pila donde se guardan los retornos de
                        subrutina */
                        funcion_interna RESET_GOSUB(;)
                        fin_funcion_interna;
                        goto MENU_PRINCIPAL: ;
        fin_elegir_opcion;
        retornar;
FIN_SUBRUTINA
```

```
SECCION_APLICACION:
        interrupcion = 1;
        reintentos = 3;
        timeout = 5;
        set_reconocimiento v;
INICIAR:
        esperar_llamada;
        descolgar;
        sintetizar "Bienvenido al buzón telefónico del grupo de tecnologías del
        habla";

        elegir_opcion
        opciones_activas = "SN";
        sintetizar elegir.txt;
                case "NO":
                        break;
                case "SI":
                        set_reconocimiento t;
                        break;
        fin_elegir_opcion;
        sintetizar "A continuación se presenta el menú principal.";

MENU_PRINCIPAL:
        elegir_opcion
        opciones_activas = "123";
        sintetizar menu_ppa.txt;
                case "uno"  :
                        goto RECIBIR_MENSAJES;
                case "dos"  :
                        goto DEJAR_MENSAJES;
                case "tres" :
                        goto DESPEDIDA;
        fin_elegir_opcion;
```

```
SECCION_APLICACION:
INICIO:
        n_interrumpir = 1;                  /* 1=SI, 0=NO */
        n_max_tiempo_reco_palabra = 5;      /* en segundos */
        s_tipo_reco="ambos";                /* tipo reconocimiento a utilizar */
        n_timeout_sistema = 40;             /* máxima duración llamada, minutos */
        n_MAX_NUM_INTENTOS = 4;             /* intentos,  no reintentos */
ESPERAR:
        n_segunda_vez = 0;                  /* para verificar cadenas */
        esperar_llamada(;);
        sintetizar("Bienvenido al buzón telefónico del grupo de tecnologías del
        habla.";);
        elegir_opcion
        opciones_activas = "SN";
        sintetizar(“Diga sí si desea utilizar reconocimiento de tonos multifrecuencia y
        no si desea utilizar reconocimiento de voz.";);
        switch
                case "NO":
                        break;
                case "SI":
                        s_tipo_reco = "DTMF";
                        break;
        fin_elegir_opcion;
        sintetizar("A continuación se presenta el menú principal.";);
MENU_PRINCIPAL:
        elegir_opcion
        opciones_activas = "123";
        sintetizar ("1 para recibir mensajes, 2 para dejar mensajes, 3 para salir.";);
        switch
                case "1" :
                        goto RECIBIR_MENSAJES: ;
                case "2" :
                        goto DEJAR_MENSAJES: ;
                case "3" :
                        goto DESPEDIDA: ;
        fin_elegir_opcion;
```

```
DEJAR_MENSAJES:
        reconocer_cadena
                1,7 digitos;
                asignar_a CODIGO;
                sintetizar "Introduzca el código personal del destinatario.";
        fin_reconocer_cadena;

        funcion_interna Inicializar_Base("buzon","access");
                case –1: case -2: case –3: case -4:
                case -5:
                        gosub FALLO_SISTEMA;
                        break;
         fin_funcion_interna;
CHEQUEAR:
         funcion_interna Montar_Query("3","select \"direccion\" from usuarios where
        \"codigo\" = '",CODIGO,"';query);
                case -1:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;

        funcion_interna Buscar_Elemento("buzon","access",query);
                case 1:
                        goto NO_HAY_CODIGO;
                case 0:
                        break;
                case -2:
                case -5:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;

        funcion_interna Leer_Campos(;MAIL);
                case 1:
                         goto NO_HAY_CODIGO;
                case 0:
                        goto DEJAR_MENSAJE;
                case -1: case -5:
                case -2:
                         gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;
```

```
DEJAR_MENSAJES:
        reconocer_cadena
                1,7 digitos;
                asignar_a s_codigo;
                asignar_a s_codigo_segundo;
                sintetizar("Introduzca el código personal del destinatario.";);
        fin_reconocer_cadena;

        ODBC_ABRE_BD("buzon","access","0";);
        funcion_interna RESULTADO_ANTERIOR(;)
        switch
                case "-1" :case "-2" : case "-3" : case "-4" :
                case "-5" :
                        gosub FALLO_SISTEMA:;
                        break;
        fin_funcion_interna;
CHEQUEAR:
        s_trozo1="select \"direccion\" from usuarios where \"codigo\" = '";
        STRCAT(s_trozo1,s_codigo,"';s_query);

        ODBC_BUSCAR("buzon","access",s_query;);
        funcion_interna RESULTADO_ANTERIOR(;)
        switch
                case "1":  goto NO_HAY_CODIGO: ;
                case "0": break;
                case "-2" :
                case "-5" :  gosub FALLO_SISTEMA:;
                        break;
        fin_funcion_interna;

        ODBC_LEER_CAMPOS(;s_mail);
        funcion_interna RESULTADO_ANTERIOR(;)
        switch
                case "1": goto NO_HAY_CODIGO: ;
                case "0": goto DEJAR_MENSAJE: ;
                case "-1" : case "-2" :
                case "-5" :
                        gosub FALLO_SISTEMA:;
                        break;
        fin_funcion_interna;
```

```
NO_HAY_CODIGO:
        sintetizar "Su código es %.",CODIGO;

        elegir_opcion
        opciones_activas = "SN";
        sintetizar "Diga sí si el código es correcto, y no en caso contrario.";
                case "no" :
                        goto PEDIR_VERIFICACION_CODIGO;
                case "si" :
                        sintetizar "El código especificado no es válido.";
                        goto MENU_PRINCIPAL;
        fin_elegir_opcion;
PEDIR_VERIFICACION_CODIGO:
        funcion_interna SegundaVez (0, "dni_es.mst", 1, CODIGO);
                case  0 :
                        goto OPCION1;
                case  1 :
                        goto OPCION2;
        fin_funcion_interna;
OPCION1:
        elegir_opcion
        opciones_activas = "SCA";
        sintetizar "Usted puede cambiar algún dígito diciendo sí, introducir de
        nuevo el código diciendo cancelar, o pedir ayuda diciendo ayuda.";
                case "cancelar" :
                        gosub PEDIR_CODIGO;
                        goto CHEQUEAR;
                case "ayuda"    :
                        gosub AYUDA_VERIFICACION;
                        goto PEDIR_VERIFICACION_CODIGO;
                case "si"  :
                        goto VERIFICAR_DNI_MEN2;
        fin_elegir_opcion;
```

```
NO_HAY_CODIGO:
        SPRINTF(s_sinte,"Su código es %s.",s_codigo;);
        sintetizar(s_sinte;);

        elegir_opcion
        opciones_activas = "SN";
        sintetizar("Diga sí si el código es correcto y no en caso contrario.";);
        switch
                case "no" :
                        goto PEDIR_VERIFICACION_CODIGO: ;
                case "si" :
                        sintetizar ("El código especificado no es válido.";);
                        goto MENU_PRINCIPAL: ;
          fin_elegir_opcion;
PEDIR_VERIFICACION_CODIGO:
        if (n_segunda_vez == 0)
                n_segunda_vez++;

                elegir_opcion
                opciones_activas = "SCA";
                sintetizar("Usted puede cambiar algun dígito diciendo si,
                introducir de nuevo el código diciendo cancelar, o pedir ayuda
                diciendo ayuda.";);
                switch
                        case "cancelar" :
                                gosub PEDIR_CODIGO: ;
                                goto CHEQUEAR: ;
                        case "ayuda"    :
                                gosub AYUDA_VERIFICACION: ;
                                goto PEDIR_VERIFICACION_CODIGO: ;
                        case "si"       :
                                goto VERIFICAR_DNI_MEN2: ;
                fin_elegir_opcion;
        endif
```

```
OPCION2:
        elegir_opcion
        opciones_activas = "SC";
        reproducir vernis2.mst;
                case "cancelar" :
                        gosub PEDIR_CODIGO;
                        goto CHEQUEAR;
                case "si"    :
                        goto VERIFICAR_DNI_MEN2;
        fin_elegir_opcion;


VERIFICAR_DNI_MEN2:
        sintetizar "Vamos a comprobar que su código es correcto.";

        reproducir verifi.mst;

        funcion_interna Verificar_Cadena (CODIGO; CODIGO);
                case 0 :
                case 1 :
                        break;
                case -1:
                case -2:
                case -3:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;
```

```
        if (n_segunda_vez == 1)   /* inncesario, pero mas legible */
                elegir_opcion
                opciones_activas = "SC";
                reproducir ("vernis2";);
                switch
                        case "cancelar" :
                                gosub PEDIR_CODIGO: ;
                                goto CHEQUEAR: ;
                        case "si"    :
                                goto VERIFICAR_DNI_MEN2: ;
                fin_elegir_opcion;
        endif


VERIFICAR_DNI_MEN2:
        sintetizar("Vamos a comprobar que su código es correcto.";);

        reproducir ("verifi";);

        s_cadena = s_codigo;
        s_segundo = s_codigo_segundo;
        gosub VerifCadena:;
        s_codigo = s_cadena;
        s_codigo_segundo = s_segundo;
```

```
COMPROBACION:
        sintetizar "Espere un momento, por favor.";

        funcion_interna Montar_Query ("3","select \"direccion\" from usuarios
        where \"codigo\" = '",CODIGO,"';query);
                case -1:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;

        funcion_interna Buscar_Elemento("buzon","access",query);
                case 1:
                        sintetizar "Lo sentimos, pero el codigo no es válido.";
                        goto DESPEDIDA;
                case 0:
                        break;
                case -2:
                case -5:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;

        funcion_interna Leer_Campos(;MAIL);
                case 1:
                        sintetizar "Lo sentimos, pero el código no es válido.";
                        goto DESPEDIDA;
                case 0:
                        goto DEJAR_MENSAJE;
                case -1:
                case -5:
                case -2:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;
```

```
COMPROBACION:
        sintetizar("Espere un momento por favor.";);

        s_trozo1="select \"direccion\" from usuarios where \"codigo\" = '";
        STRCAT(s_trozo1,s_codigo,"';s_query);

        ODBC_BUSCAR("buzon","access",s_query;);
        funcion_interna RESULTADO_ANTERIOR(;)
        switch
                case "1":
                        sintetizar ("Lo sentimos, pero el código no es válido.";);
                        goto DESPEDIDA: ;
                case "0":
                        break;
                case "-2" :
                case "-5" :
                        gosub FALLO_SISTEMA:;
                        break;
        fin_funcion_interna;

        ODBC_LEER_CAMPOS(;s_mail);
        funcion_interna RESULTADO_ANTERIOR(;)
        switch
                case "1":
                        sintetizar ("Lo sentimos, pero el código no es válido.";);
                        goto DESPEDIDA: ;
                case "0":
                        goto DEJAR_MENSAJE: ;
                case "-1" :
                case "-2" :
                case "-5" :
                        gosub FALLO_SISTEMA:;
                        break;
        fin_funcion_interna;
```

```
DEJAR_MENSAJE:
        sintetizar "Deje el mensaje después de escuchar el pitido.";

        gosub ALMACENAR;

        funcion_interna Enviar_Mail(MAIL,"new_mens.txt");
        fin_funcion_interna;
        goto MAS_OPERACIONES;

RECIBIR_MENSAJES:
        reconocer_cadena
                7 digitos;
                asignar_a CODIGO;
                sintetizar "Introduzca su código personal.";
        fin_reconocer_cadena;

INTRODUCIR_CLAVE:
          reconocer_cadena
                  4 digitos;
                  asignar_a CLAVE;
                  sintetizar "Introduzca su clave.";
          fin_reconocer_cadena;

CHEQUEAR1:
        funcion_interna Inicializar_Base("buzon","access");
                case -1:  case -2:  case -3:  case -4:
                case -5:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;

        funcion_interna Montar_Query("5","select \"codigo\" from usuarios where
        \"codigo\" = '",CODIGO,"' and \"clave\" = '",CLAVE,"';query);
                case -1:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;
```

```
DEJAR_MENSAJE:
        sintetizar("Deje el mensaje después de escuchar el pitido.";);

        gosub ALMACENAR: ;

        ENVIAR_MAIL(s_mail,"new_mens.txt";);
        goto MAS_OPERACIONES: ;
RECIBIR_MENSAJES:
        reconocer_cadena
                1,7 digitos;
                asignar_a s_codigo;
                asignar_a s_codigo_segundo;
                sintetizar("Introduzca su código personal.";);
        fin_reconocer_cadena;
INTRODUCIR_CLAVE:
        reconocer_cadena
                1,4 digitos;
                asignar_a s_clave;
                asignar_a s_clave_segunda;
                sintetizar("Introduzca su clave.";);
        fin_reconocer_cadena;

CHEQUEAR1:
        ODBC_ABRE_BD("buzon","access","0";);
        funcion_interna RESULTADO_ANTERIOR(;)
        switch
                case "-1" : case "-2" : case "-3" : case "-4" :
                case "-5" :
                        gosub FALLO_SISTEMA:;
                        break;
        fin_funcion_interna;

        STRCAT(“select \"codigo\" from usuarios where \"codigo\" = '”,
        s_codigo,"' and \"clave\" = '",s_clave,"';s_query);
```

```
funcion_interna Buscar_Elemento("buzon","access",query);
        case 1:
                goto NO_HAY_CODIGO1;
        case 0:
                break;
        case -2:
        case -5:
                gosub FALLO_SISTEMA;
                break;
fin_funcion_interna;


funcion_interna Leer_Campos(;CODIGO1);
        case 1:
                goto NO_HAY_CODIGO1;
        case 0:
                goto DA_MENSAJES;
        case -1:
        case -2:
        case -5:
                gosub FALLO_SISTEMA;
                break;
fin_funcion_interna;
```

```
ODBC_BUSCAR("buzon","access",s_query;);
funcion_interna RESULTADO_ANTERIOR(;)
switch
        case "1":
                goto NO_HAY_CODIGO1: ;
        case "0":
                break;
        case "-2" :
        case "-5" :
                gosub FALLO_SISTEMA:;
                break;
fin_funcion_interna;


ODBC_LEER_CAMPOS(;s_codigo1);
funcion_interna RESULTADO_ANTERIOR(;)
switch
        case "1":
                goto NO_HAY_CODIGO1: ;
        case "0":
                goto DA_MENSAJES: ;
        case "-1" :
        case "-2" :
        case "-5" :
                gosub FALLO_SISTEMA:;
                break;
fin_funcion_interna;
```

```
NO_HAY_CODIGO1:
        sintetizar "Su código es %.",CODIGO;

        elegir_opcion
        opciones_activas = "SN";
        sintetizar "¿Es correcto?";
                case "no" :
                        goto PEDIR_VERIFICACION_CODIGO1;
                case "si" :
                        goto NO_HAY_CLAVE;
        fin_elegir_opcion;

PEDIR_VERIFICACION_CODIGO1:
        funcion_interna SegundaVez (0, "dni_es.mst", 1, CODIGO);
                case  0 :
                        goto OPCION1_RECIBIR;
                case  1 :
                        goto OPCION2_RECIBIR;
        fin_funcion_interna;

OPCION1_RECIBIR:
        elegir_opcion
        opciones_activas = "SCA";
        sintetizar "Usted puede cambiar algún dígito diciendo sí, introducir de
        nuevo el código diciendo cancelar, o pedir ayuda diciendo ayuda.";
                case "cancelar" :
                        gosub PEDIR_CODIGO;
                        goto NO_HAY_CODIGO1;
                        break;
                case "ayuda" :
                        gosub AYUDA_VERIFICACION;
                        goto PEDIR_VERIFICACION_CODIGO1;
                        break;
                case "si" :
                        goto VERIFICAR_CODIGO1;
        fin_elegir_opcion;
```

```
NO_HAY_CODIGO1:
        SPRINTF(s_sinte,"Su codigo es %s.",s_codigo;);
        sintetizar(s_sinte;);

        elegir_opcion
        opciones_activas = "SN";
        sintetizar("¿ Es correcto ?";);
        switch
                case "no" :
                        goto PEDIR_VERIFICACION_CODIGO1: ;
                case "si" :
                        goto NO_HAY_CLAVE: ;
        fin_elegir_opcion;

PEDIR_VERIFICACION_CODIGO1:
        if (n_segunda_vez == 0)
                n_segunda_vez++;

                elegir_opcion
                opciones_activas = "SCA";
                sintetizar("Usted puede cambiar algún dígito diciendo si,  introducir
                de nuevo el código diciendo cancelar, o pedir ayuda diciendo
                ayuda.";);
                switch
                        case "cancelar" :
                                gosub PEDIR_CODIGO: ;
                                goto NO_HAY_CODIGO1: ;
                        case "ayuda"    :
                                gosub AYUDA_VERIFICACION: ;
                                goto PEDIR_VERIFICACION_CODIGO1: ;
                        case "si"       :
                                goto VERIFICAR_CODIGO1: ;
                fin_elegir_opcion;
        endif
```

```
OPCION2_RECIBIR:
        elegir_opcion
        opciones_activas = "SC";
        reproducir vernis2.mst;
                case "cancelar" :
                        gosub PEDIR_CODIGO;
                        goto NO_HAY_CODIGO1;
                case "si"    :
                        goto VERIFICAR_CODIGO1;
        fin_elegir_opcion;

VERIFICAR_CODIGO1:
        sintetizar "Vamos a comprobar que su código es correcto.";

        reproducir verifi.mst;

        funcion_interna Verificar_Cadena (CODIGO; CODIGO);
                case  0 :
                case  1 :
                        break;
                case -1 :
                case -2 :
                case -3 :
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;

NO_HAY_CLAVE:
        sintetizar "Su clave es %.",CLAVE;

        elegir_opcion
        opciones_activas = "SN";
        sintetizar "¿Es correcta?";
                case "no" :
                        goto PEDIR_VERIFICACION_CLAVE;
                case "si" :
                        goto SEGUNDA_COMPROBACION;
        fin_elegir_opcion;
```

```
        if (n_segunda_vez == 1)   /* inncesario, pero mas legible */
                elegir_opcion
                opciones_activas = "SC";
                reproducir ("vernis2";);
                switch
                        case "cancelar" :
                                gosub PEDIR_CODIGO: ;
                                goto NO_HAY_CODIGO1: ;
                        case "si"    :
                                goto VERIFICAR_CODIGO1: ;
                fin_elegir_opcion;
        endif

VERIFICAR_CODIGO1:
        sintetizar("Vamos a comprobar que su código es correcto.";);

        reproducir ("verifi";);

        s_cadena=s_codigo;
        s_segundo=s_codigo_segundo;
        gosub VerifCadena:;
        s_codigo = s_cadena;
        s_codigo_segundo = s_segundo;

NO_HAY_CLAVE:
        SPRINTF(s_sinte,"Su clave es %s.",s_clave;);
        sintetizar(s_sinte;);

        elegir_opcion
        opciones_activas = "SN";
        sintetizar("¿ Es correcta ?";);
        switch
                case "no" :
                        goto PEDIR_VERIFICACION_CLAVE: ;
                case "si" :
                        goto SEGUNDA_COMPROBACION: ;
        fin_elegir_opcion;
```

```
PEDIR_VERIFICACION_CLAVE:
        funcion_interna SegundaVez (0, "pass_es.mst", 1, CLAVE);
                case  0 :
                        goto OPCION3;
                case  1 :
                        goto OPCION4;
        fin_funcion_interna;
OPCION3:
        elegir_opcion
        opciones_activas = "SCA";
        sintetizar "Usted puede cambiar algún dígito diciendo sí, introducir de
        nuevo la clave diciendo cancelar, o pedir ayuda diciendo ayuda.";
                case "cancelar" :
                        gosub PEDIR_CLAVE;
                        goto COM_CLAVE;
                case "ayuda"    :
                        gosub AYUDA_VERIFICACION;
                        goto PEDIR_VERIFICACION_CLAVE;
                case "si"       :
                        goto VERIFICAR_CLAVE;
        fin_elegir_opcion;
OPCION4:
        elegir_opcion
        opciones_activas = "SC";
        reproducir verpas2.mst;
                case "cancelar" :
                        gosub PEDIR_CLAVE;
                        goto COM_CLAVE;
                case "si”    :
                        goto VERIFICAR_CLAVE;
        fin_elegir_opcion;
```

```
PEDIR_VERIFICACION_CLAVE:
        if (n_segunda_vez == 0)
                n_segunda_vez++;

                elegir_opcion
                opciones_activas = "SCA";
                sintetizar("Usted puede cambiar algun dígito diciendo si, introducir
                de nuevo el código diciendo cancelar, o pedir ayuda diciendo
                ayuda.";);
                switch
                        case "cancelar" :
                                gosub PEDIR_CLAVE: ;
                                goto COM_CLAVE: ;
                        case "ayuda"    :
                                gosub AYUDA_VERIFICACION: ;
                                goto PEDIR_VERIFICACION_CLAVE: ;
                        case "si"       :
                                goto VERIFICAR_CLAVE: ;
                fin_elegir_opcion;
        endif
        if (n_segunda_vez == 1)   /* inncesario, pero mas legible */
                elegir_opcion
                opciones_activas = "SC";
                reproducir ("verpas2";);
                switch
                        case "cancelar" :
                                gosub PEDIR_CLAVE: ;
                                goto COM_CLAVE: ;
                        case "si"    :
                                goto VERIFICAR_CLAVE: ;
                fin_elegir_opcion;
        endif
```

```
VERIFICAR_CLAVE:
        reproducir verifi.mst;
        funcion_interna Verificar_Cadena (CLAVE; CLAVE);
                case  0 :
                case  1 : break;
                case -1 :
                case -2 :
                case -3 : gosub FALLO_SISTEMA;
                          break;
        fin_funcion_interna;
        goto SEGUNDA_COMPROBACION;
COM_CLAVE:
        sintetizar "Su clave es %.",CLAVE;

        elegir_opcion
        opciones_activas = "SN";
        sintetizar "¿Es correcta?";
                case "no" :
                        goto PEDIR_VERIFICACION_CLAVE;
                case "si" :
                        goto SEGUNDA_COMPROBACIION;
        fin_elegir_opcion;
SEGUNDA_COMPROBACION:
        reproducir espere.mst;

        funcion_interna Montar_Query("5","select \"codigo\" from usuarios where
        \"codigo\" = '",CODIGO,"' and \"clave\" = '",CLAVE,"';query);
                case -1:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;

        funcion_interna Buscar_Elemento("buzon","access",query);
                case 1:
                        sintetizar "Lo sentimos, pero el código o la clave no son
                        válidos.";
                        goto DESPEDIDA;
                case 0:   break;
                case -2:
                case -5:  gosub FALLO_SISTEMA;
                        Break;
        fin_funcion_interna;
```

```
VERIFICAR_CLAVE:
        reproducir ("verifi";);
        s_cadena = s_clave;
        s_segundo = s_clave_segundo;
        gosub VerifCadena:;
        s_clave = s_cadena;
        s_clave_segundo = s_segundo;
        goto SEGUNDA_COMPROBACION: ;
COM_CLAVE:
        SPRINTF(s_sinte,"Su clave es %s.",s_clave;);
        sintetizar(s_sinte;);

        elegir_opcion
        opciones_activas = "SN";
        sintetizar("¿ Es correcta ?";);
        switch
                case "no" :
                        goto PEDIR_VERIFICACION_CLAVE: ;
                case "si" :
                        goto SEGUNDA_COMPROBACION: ;
        fin_elegir_opcion;
SEGUNDA_COMPROBACION:
        reproducir ("espere";);

        s_trozo1="select \"codigo\" from usuarios where \"codigo\" = '";
        STRCAT(s_trozo1,s_codigo,"' and \"clave\" = '",s_clave,"';s_query);

        ODBC_BUSCAR("buzon","access",s_query;);
        funcion_interna RESULTADO_ANTERIOR(;)
        switch
                case "1":
                        sintetizar ("Lo sentimos, pero el código o la clave no son
                        válidos.";);
                        goto DESPEDIDA: ;
                case "0":
                        break;
                case "-2" :
                case "-5" :
                        gosub FALLO_SISTEMA:;
                        break;
        fin_funcion_interna;
```

```
        funcion_interna Leer_Campos(;CODIGO1);
                case 1:
                        sintetizar "Lo sentimos, pero el código o la clave no son
                        válidos.";
                        goto DESPEDIDA;
                case 0:  goto DA_MENSAJES;
                case -1:
                case -2:
                case -5:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;

DA_MENSAJES:
        sintetizar "A continuación se presentan los mensajes que se han recibido
        desde su última llamada.";

        funcion_interna Montar_Query("3","select \"fichero\" from mensaje where
        \"codigo\" = '",CODIGO,"' and \"escuchado\" = 'N'";query);
                case -1 :
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;

        funcion_interna Buscar_Elemento ("buzon","access",query);
                case 1 :
                        sintetizar "No tiene ningún mensaje nuevo.";
                        goto MAS_OPERACIONES;
                case 0 :  break;
                case -5 :
                case -2 :
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;
```

```
        ODBC_LEER_CAMPOS(;s_codigo1);
        funcion_interna RESULTADO_ANTERIOR(;)
        switch
                case "1":
                        sintetizar ("Lo sentimos, pero el código o la clave no son
                        válidos.";);
                        goto DESPEDIDA: ;
                case "0":
                        goto DA_MENSAJES: ;
                case "-1" :
                case "-2" :
                case "-5" :
                        gosub FALLO_SISTEMA:;
                        break;
        fin_funcion_interna;

DA_MENSAJES:
        sintetizar("A continuación se presentan los mensajes que se han recibido
        desde su última llamada.";);

        s_trozo1=;
        STRCAT("select \"fichero\" from mensaje where \"codigo\" = '",
        s_codigo,"' and \"escuchado\" = 'N'";s_query);
        ODBC_BUSCAR ("buzon","access",s_query;);
        funcion_interna RESULTADO_ANTERIOR(;)
        switch
                case "1" :
                        sintetizar ("No tiene ningún mensaje nuevo.";);
                        goto MAS_OPERACIONES: ;
                case "0" :
                        break;
                case "-2" :
                case "-5" :
                        gosub FALLO_SISTEMA:;
                        break;
        fin_funcion_interna;
```

```
REPRODUCIR_MENSAJE:
        funcion_interna Leer_Campos (;fichero);
                case  0 :
                        pitar;

                        reproducir * fichero;

                        goto REPRODUCIR_MENSAJE;
                case 1:
                        sintetizar "No hay más mensajes.";
                        break;
                case -1 :
                case -2 :
                case -5 :
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;


ETIQUETAR_MENSAJE:
        funcion_interna Montar_Query("3","update mensaje set \"escuchado\" = 'S'
        where \"escuchado\" = 'N' and \"codigo\" = '",CODIGO,"'";query);
        fin_funcion_interna;

        funcion_interna Buscar_Elemento("buzon","access",query);
                case 1:
                case 0:
                        break;
                case -2:
                case -5:
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;

        goto MENU2;
```

```
REPRODUCIR_MENSAJE:
        ODBC_LEER_CAMPOS (;s_fichero);
        funcion_interna RESULTADO_ANTERIOR(;)
        switch
                case  "0" :
                        pitar(;);

                        /* hay que cambiar el directorio de reproduccion */
                        s_dir_anterior = s_DIR_ESPECIF;
                        s_DIR_ESPECIF = ".\\grabacio\\";
                        reproducir (s_fichero;);

                        goto REPRODUCIR_MENSAJE: ;
                case "1":
                        sintetizar ("No hay más mensajes.";);
                        break;
                case "-1" : case "-2" : case "-5" :
                        gosub FALLO_SISTEMA:;
                        break;
        fin_funcion_interna;

        /* dejamos el directorio de reproducción como estaba */
        s_DIR_ESPECIF = s_dir_anterior;
ETIQUETAR_MENSAJE:
        s_trozo1="update mensaje set \"escuchado\" = 'S' where \"escuchado\" = 'N'
        and \"codigo\" = '";
        STRCAT(s_trozo1,s_codigo,"'";s_query);

        ODBC_BUSCAR("buzon","access",s_query;);
        funcion_interna RESULTADO_ANTERIOR(;)
        switch
                case "1":
                case "0":
                        break;
                case "-2" :
                case "-5" :
                        gosub FALLO_SISTEMA:;
                        break;
        fin_funcion_interna;
        goto MENU2: ;
```

```
MENU2:
        elegir_opcion
        opciones_activas = "123";
        sintetizar "Diga 1 para revisar los mensajes del buzón, 2 para cambiar su
        clave, y 3 para salir.";
                case "uno":
                        goto REVISAR_MENSAJES;
                case "dos":
                        goto CAMBIAR_CLAVE;
                case "tres":
                        goto DESPEDIDA;
        fin_elegir_opcion;

REVISAR_MENSAJES:
        sintetizar "A continuación se presentan todos los mensajes almacenados en
        su buzón.";

        funcion_interna Montar_Query("3","select \"fichero\" from mensaje where
        \"codigo\" = '",CODIGO,"';query);
                case -1 :
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;

        funcion_interna Buscar_Elemento ("buzon","access",query);
                case 1 :
                        sintetizar "No tiene ningún mensaje.";
                        goto MAS_OPERACIONES;
                case 0 :
                        break;
                case -2 :
                case -5 :
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;
```

```
MENU2:
        elegir_opcion
        opciones_activas = "123";
        sintetizar("Diga 1 para revisar los mensajes del buzón, 2 para cambiar su
        clave, y 3 para salir.";);
        switch
                case "1":
                        goto REVISAR_MENSAJES: ;
                case "2":
                        goto CAMBIAR_CLAVE: ;
                case "3":
                        goto DESPEDIDA: ;
        fin_elegir_opcion;

REVISAR_MENSAJES:
        sintetizar("A continuación se presentan todos los mensajes almacenados en
        su buzón.";);

        s_trozo1="select \"fichero\" from mensaje where \"codigo\" = '";
        STRCAT(s_trozo1,s_codigo,"';s_query);

        ODBC_BUSCAR ("buzon","access",s_query;);
        funcion_interna RESULTADO_ANTERIOR(;)
        switch
                case "1" :
                        sintetizar ("No tiene ningún mensaje.";);
                        goto MAS_OPERACIONES: ;
                case "0" :
                        break;
                case "-2" :
                case "-5" :
                        gosub FALLO_SISTEMA:;
                        break;
        fin_funcion_interna;
```

```
REPRODUCIR_TODOS_MENSAJES:
        funcion_interna Leer_Campos(;fichero);
                case  0 :
                        pitar;

                        reproducir * fichero;

                        gosub BORRAR;
                        goto REPRODUCIR_TODOS_MENSAJES;
                        break;
                case 1:
                        sintetizar "No hay más mensajes.";
                        break;
                case -1 :
                case -5 :
                case -2 :
                        gosub FALLO_SISTEMA;
                        break;
        fin_funcion_interna;
        goto MENU2;
```

```
REPRODUCIR_TODOS_MENSAJES:
        ODBC_LEER_CAMPOS (;s_fichero);
        funcion_interna RESULTADO_ANTERIOR(;)
        switch
                case  "0" :
                        pitar(;);

                        /* hay que cambiar el directorio de reproduccion */
                        s_dir_anterior = s_DIR_ESPECIF;
                        s_DIR_ESPECIF = ".\\grabacio\\";
                        reproducir (s_fichero;);

                        gosub BORRAR: ;
                        goto REPRODUCIR_TODOS_MENSAJES: ;
                        break;
                case "1":
                        sintetizar ("No hay más mensajes.";);
                        break;
                case "-1" :
                case "-2" :
                case "-5" :
                        gosub FALLO_SISTEMA:;
                        break;
        fin_funcion_interna;

        /* dejamos el directorio de reproducción como estaba */
        s_DIR_ESPECIF = s_dir_anterior;

        goto MENU2: ;
```

```
CAMBIAR_CLAVE:
        reconocer_cadena
                4 digitos;
                asignar_a CLAVE_NEW;
                sintetizar "Introduzca su nueva clave.";
        fin_reconocer_cadena;

        elegir_opcion
        opciones_activas = "SN";
        sintetizar "¿Es la clave introducida %?", CLAVE_NEW;
                case "si" :
                        gosub CAMBIO_CLAVE;
                        break;
                case "no" :
                        elegir_opcion
                        opciones_activas = "SN";
                        sintetizar "¿Quieres volver a intentar introducir la
                        clave?";
                                case "si" :  goto CAMBIAR_CLAVE;
                                case "no" : goto MENU_PRINCIPAL;
                        fin_elegir_opcion;
                        break;
        fin_elegir_opcion;
        goto MENU2;

MAS_OPERACIONES:
        elegir_opcion
        opciones_activas = "SN";
        sintetizar "¿Desea efectuar alguna operación más?";
                case "si" :
                        goto MENU_PRINCIPAL;
                case "no" :
                        goto DESPEDIDA;
        fin_elegir_opcion;

DESPEDIDA:
        sintetizar adios.txt;
        colgar;
        funcion_interna Cerrar_Base ("buzon","access");
        fin_funcion_interna;
        reiniciar;
FIN:
```

```
CAMBIAR_CLAVE:
        reconocer_cadena
                1,4 digitos;
                asignar_a s_clave_new;
                asignar_a s_clave_new_segunda;
                sintetizar("Introduzca su nueva clave.";);
        fin_reconocer_cadena;

        SPRINTF(s_sinte,"¿ Es la clave introducida %s ?.",s_clave_new;);
        elegir_opcion
        opciones_activas = "SN";
        sintetizar(s_sinte;);
        switch
                case "si" :
                        gosub CAMBIO_CLAVE: ;
                        break;
                case "no" :
                        /* como no no es posible anidar dos instrucciones del
                        tipo elegir_opcion, llamamos a una subrutina */
                        gosub SUB1: ;
                        break;
        fin_elegir_opcion;
        goto MENU2: ;

MAS_OPERACIONES:
        elegir_opcion
        opciones_activas = "SN";
        sintetizar("¿ Desea efectuar alguna operación más ?";);
        switch
                case "si" :
                        goto MENU_PRINCIPAL: ;
                case "no" :
                        goto DESPEDIDA: ;
        fin_elegir_opcion;

DESPEDIDA:
        sintetizar ("Gracias por haber utilizado nuestro servicio automático.
        Hasta la próxima vez.";);
        colgar(;);
        ODBC_CIERRA_BD ("buzon","access";);
FIN:
```

APÉNDICE C.COMPARACIÓN ENTRE EL LENGUAJE DEL SISTEMA MONOLÍNEA Y EL LENGUAJE DEL SISTEMA MULTILÍNEA *1*

C.1 INTRODUCCIÓN **1**

C.2 DESCRIPCIÓN DEL SERVICIO DE BUZÓN TELEFÓNICO **2**

C.2.1 INTRODUCCIÓN 2

C.2.2 FUNCIONAMIENTO 2

C.3 DIFERENCIAS ENTRE AMBOS LENGUAJES **5**

C.3.1 ASPECTOS GENERALES 5

C.3.2 SECCIÓN ERRORES 5

C.3.3 SECCIÓN SUBRUTINAS 5

C.3.4 SECCIÓN APLICACIÓN 6

C.3.5 OTRAS CONSIDERACIONES 8

COMPARACIÓN ENTRE BUZON.LDD Y BUZON.PRO 9